ISBN 978-981-14-1684-2
Proceedings of 2019 the 9th International Workshop on Computer Science and Engineering
(WCSE 2019 SUMMER)
Hong Kong, 15-17 June, 2019, pp. 715-720
doi: 10.18178/wcse.2019.06.106

Research on Network Public Opinion Detection Based on Improved TF-IDF Algorithm

Lu Peng[1+], Zongfeng Qin [2]

[1, 2] City College, Wuhan University of Science and Technology

Abstract. TF-IDF algorithm is a widely used text feature weighting technology. The core idea of TF-IDF algorithm is as follows: In a corpus, if a participle appears frequently in a certain text and appears less in other texts, then it proves that the participle has a good feature of expression to this text. Although this idea is very simple, it also faces some problems in practical applications. Because it blindly increased the importance of uncommon words in the text and this blindness will also appear in the field of public opinion monitoring. In order to solve the mentioned problem, this thesis has done the following work:

- Introduce the lexical weight coefficient of the characteristic word into TF-IDF;
- Introduce the word position weight (span weight) coefficient into TF-IDF.

The experiment proves that the improved TF-IDF method highlights the importance of text feature words and facilitates classification. Furthermore, the improved method is applied to the public opinion analysis system and got good results.

Keywords: Network Public Opinion; Cosine Similarity; TF-IDF; Emotional Analysis.

1. Introduction

In information retrieval, TF-IDF is the abbreviation of Term Frequency-Inverse Document Frequency. It is widely used in the field of information retrieval and data mining as a simple and efficient word weighting method. The two-part composition, TF is the importance of a word as it increases in the number of occurrences in the text, IDF indicates that the word decreases as it appears in the text set in the text. Today, TF-IDF is one of the most popular terminology weighting schemes.

1.1. Principle introduction

TF-IDF is essentially the product of TF and IDF. The word frequency TF calculation formula is given by formula (1), which indicates the frequency at which the word is the characteristic item appearing in the single document. For a word, the TF calculation formula is as follows[1]:

$$\mathrm{TF}_{i,j} = \frac{N_{i,j}}{\sum_{k=0}^{n} N_{k,j}} \tag{1}$$

where $N_{i,j}$ represents the number of occurrences of the ith word in the document D_j, and the denominator represents the total number of words in the document d_j.

For the inverse document frequency IDF calculation formula[1], see (2).

$$\mathrm{IDF} = \log\left(\frac{|D|}{|\{d_j \in D: t_i \in d_j\}|}\right) \tag{2}$$

In this formula:

- |D|:D is the total number of documents in the corpus
- |{dj∈D: ti∈dj}|: Number of documents containing feature items ti in the corpus

The calculation formula (3)[1] of TF-IDF can be obtained by combining formula (1) and formula (2):

+ Corresponding author. Tel.: 1 (13971055934)
E-mail address: Jasmin001peng@gmail.com

$$\text{TF-IDF=TF*IDF=}\frac{N_{i,j}}{\sum_{k=0}^{n} N_{k,j}} * \log\left(\frac{|D|}{|\{d_j \in D: t_i \in d_j\}|}\right) \quad (3)$$

According to the formula, the more the number of corpus documents belonging to the feature item, the smaller the IDF result, that is, the wider the range of the feature item is involved, the worse the text distinguishing ability of the feature item, and when a feature item is included in each document of the corpus, the resulting feature weight is 0. In order to avoid the case where the denominator is 0 when there is no feature in all documents, many practical applications use a method of adding 1 to the denominator.

1.2. TF-IDF application

TF-IDF algorithm is widely used in many fields, mainly including search engine, automatic extraction of feature words and automatic summary[2-3].

- Search engines: We typically use an improved TF-IDF weighting scheme to evaluate user query terms and document relevance, providing users with optimal matching results;
- Automatic extraction of feature words : It is widely used in finding keywords or the words that you need, as it's easy to clarify the keywords in a document;
- Automatic Summary: In many websites, such as essay websites, news websites, blogs, and search engines, in order to facilitate users to find their interesting topics, a summary of hotspots is provided for users to choose.

1.3. Insufficiency of TF-IDF

According to the above description of the related concept of TF-IDF, the importance of a word is mainly measured by the frequency of words. However it is obviously lack of theoretical basis that thinking high-frequency words are not important while low-frequency words are important in a document. Because common words do not represent meaningless and low-frequency words do not necessarily have good feature expression ability[4]. Neither the position of the words appearing nor the influence of part-of-speech of words on feature extraction are well reflected in the algorithm.

In the text collection, the total number of documents D is constant, the feature word M_1 appears n_1 times in category C, and n_2 times in other categories of text collection, when $n_1 >> n_2$, according to formula (2), the IDF calculation results of M_1 for C are small. Analyzing the distribution of M_1 in the text set which actually has a better classification effect. In addition, when the following table appears:

Table 1: Distribution Characteristics between Words

Document category / words	M_1	M_2
C_1	9	5
C_2	1	5

Note: The number of documents in categories C_1 and C_2 is 10.

In order to highlight the influence of the distribution difference of feature words between document categories on the IDF calculation, the value of the numerator D in the formula (2) is replaced with the total number of documents containing the feature words in the corpus. The values of the IDFs of the feature items M_1 and M_2 in the two types of documents are calculated from the above table data:

$$IDF_{M_1}=\log\left(\frac{10}{9}\right)= 0.045757490560675 , \qquad IDF_{M_2}=\log\left(\frac{10}{5}\right)= 0.30102999566398$$

Observing the above table, we can find that M_1 is mainly concentrated in document C_1, while word M_2 is very evenly distributed in two types of text. Obviously, word M_1 has better representation than M_2, and the calculation result does not match the fact, so the calculation in IDF exists defects.

2. TF-IDF Improvements

2.1. Improve proposals

Usually different parts of the text are given different weight factors L. If a word appears in different parts of the text, its average value is taken, and when the weight of the feature word is calculated, the part of speech N can be considered. Generally speaking, the order of importance of the part of speech is: proper nouns > nouns > verbs > adjectives, adverbs > others. Therefore, in the text classification, we can take the

product TF-IDF*L_i*N_i of the TF-IDF and the feature word position weight L_i and the part-of-speech weight N_i as the weight of the feature word. We propose a solution for the deficiency of IDF. When calculating the IDF, the document probability ratio of the feature word in the text set can be used instead of the times appearing in the text set document. The detailed improvement steps are as follows:

There is a set of text to be classified S={C_1, C_2,..., C_k}, and there is a text set D={d_1, d_2,..., d_n} in the class C_i($C_i \in S$). The keyword W appears in t documents during these n documents, and in t' documents during the rest n' documents, and the probability that the keyword W appears in the class C_i is P_m: The probability of appearing in the entire set S is P_m':

$$P_m = \frac{t}{n} \quad (4) \qquad P_m' = \frac{t+t'}{n+n'} \quad (5)$$

So the IDF calculation formula as below:

$$IDF=\log(\frac{P_m}{P_m'})=\log(1+\frac{n'}{n})-\log(1+\frac{t'}{t})=\log(1+\frac{n'}{n})+\log(\frac{t}{t+t'}) \quad (6)$$

Let $A = \log(1+\frac{n'}{n})$, $B= \log(\frac{t}{t+t'})$

Analyzing the above formula, when n'>> n, the ratio of n' and n can be used to reduce the calculation error caused by the difference in the number of documents. When t>>t', the distribution of characteristic words in the category C_i is concentrated. At this time, the value of $\frac{t}{t+t'}$ is close to 1, and the absolute value of the expression B is rather small, so it needs to be inverted to meet the actual demand. According to the logarithmic function, the independent variable is a positive number. The result of B should be positive, so:

$$B=-\log(1-\frac{t}{t+t'}) = \log(1+\frac{t}{t'}) \quad (7) \qquad IDF= \log(1+\frac{n'}{n})+ \log(1+\frac{t}{t'}) \quad (8)$$

Use the improved IDF to calculate the problem we mentioned before:

IDF_{M_1} = 1.301029995664 , IDF_{M_2} = 0.60205999132796

This result is reasonable. Next we compute the weight value considering both part of speech and position of the feature word:

The part-of-speech weight N_i and the position weight L_i of the feature word are calculated through the weight coefficients listed in Table 2 and Table 3:

Table 2: Part of Speech Weight Coefficient

Part of speech	**weight coefficient (N)**
Proper noun	5
Noun (n)	4
Verb(v)	3
Adjective, adverb(adj., adv)	2
Others	1

Table 3: Location Weight Coefficient

Word position	**weight coefficient (L)**
Main title, subhead, crosshead	3
Abstract, first paragraph, last paragraph	2
Other texts	1

Finally, the improved formula can be obtained as below:

$$TF\text{-}IDF\text{-}NL = TF*IDF*N_i * L_i \quad (9)$$

For texts that do not have a standard article structure, like comment data, we can use the formula without the position weight coefficient, as below:

$$TF\text{-}IDF\text{-}N = TF*IDF*N_i \quad (10)$$

2.2. Scheme Verification

The verification data in the experiment is through web crawling technology. The 1621 pieces of news data randomly crawled on today's headline news network are divided into six categories: technology, entertainment, finance, sports, automobile, and society. The test data of every test category is extracted in a 1:1 ratio. We use Java language with IDEA environment and "Hagong Daxunfei Voice Cloud" as our word segmentation tool. The detailed verification flow chart as figure 1.

Fig. 1: Improvement Proposals Flow Chart

From the result, it can be concluded that in the feature extraction of the text, the improved TF-IDF-NL algorithm extracts the feature words with better clustering effect and better reflects the text features, which will be more helpful in the analysis of public opinion to carry out the methods based on keywords. The shortcoming is that the position weight and the part-of-speech weight coefficient of the feature words do not give a scientific basis, which is also a problem to be solved in the future.

3. Implementation Steps of the Public Opinion Analysis System

3.1. Structured the data based on improved TF-IDF algorithm

The computer can't recognize the content of the obtained source data, so we must structure the data. The more common method is to construct the feature vector model. The specific method is to use the feature evaluation function (this article uses the improved TF-IDF algorithm) to extract the text features, then construct a feature vector model, abstract the text in a scientific way, and replace the text with text features, thereby reducing the complexity of text processing[5]. We use the later method.

3.2. Emotional analysis based on the thesaurus

Sentiment analysis can be realized by artificially configuring emotional semantic lexicon and machine learning. However the artificially constructing emotional semantic lexicon is low efficiency and low scalability but high accuracy and the machine method is complexity because of the neural algorithm, so we combines machine learning and manual emotion lexicon construction methods. First, collect a large amount of public opinion data from the network, process the data, obtain the characteristic words, construct the association between these words (the multi-group method, which will be described in detail in the next section), and then combine the emotional semantic lexicon to analyze.

3.3. Multivariate analysis

A multi-tuple consists of a number of tuples, which in turn are composed of tuple members. By setting up a multivariate tuple, an effective matching method can be provided to solve different semantic problems generated by the same word in different contexts.

The first step is to create a multi-tuple, in the second step; a tuple instance is created according to the multi-group information table; then the third step is to generate a tuple emotional state analysis table; the last step introduce the tuple association diagram:

Table 4: Multi-tuple Information(step1)

No.	Multi-tuple name	Tuple
1	Health class multi-tuple	state tuple, health tuple
2	Feature class multi-tuple	Attribute tuple
3	Proprietary class multi-tuple	Brand tuples, exclusive tuples, character tuples
4	emotional class multi-tuple	Public praise tuple, attitude tuple

Table 5: Tuple Instances(step2)

No.	tuple name	tuple	class
1	state tuple	walking, exercising, rest...	v
2	Attribute tuple	Education, interest, signs…	n
3	Health tuple	relaxation, anxiety, fatigue…	Adj, v
4	Brand tuple	Xiaomi, apple, Samsung…	n
5	Proprietary tuple	China, socialism, leaders…	nh
6	……	……	……

Table 6: Analysis of the Emotional State(step3)

No.	Multi-tuple combination	Tuple member	Emotional status
1	State tuple	sports, relaxation	positive
	Health tuple	walking, fatigue	negative
2	Brand tuple	millet, cost-effective	positive
	Public praise tuple	Apple, easy to use	positive
3	Character tuple	Ma Yun, ridicule	negative
	Attitude tuple	Lei Jun, like	positive

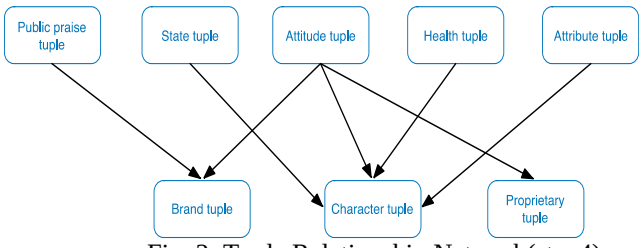

Fig. 2: Tuple Relationship Network(step4)

According to the above connection rule, when the character tuple is matched in the feature word, the emotion to be expressed can be analyzed by matching the state tuple, the attribute tuple, the health tuple or the attitude tuple.

3.4. Similarity matching analysis

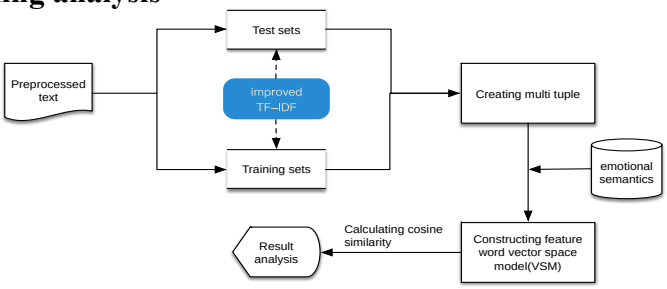

Fig. 3: Analysis of Public Opinion Flow Chart

Various new words have sprung up on the Internet. In order to reduce the influence of inaccuracy on the sentiment analysis system due to the emotional semantic lexicon, the similarity matching method will be used instead of the exact matching, and the sentiment analysis will be performed. We use cosine distance to calculate similarity matching. The process is described in Figure 3.

4. Experiment and Result Analysis

The evaluation indicators are as follows:

- The experimental environment is consistent.
- The data source is random.
- Create a comparison group

It uses the international standard to analyze the improvement rate of the improvement scheme, the recall rate, precision rate and the evaluation multi-directional verification improvement algorithm.

4.1. Experimental verification

The data used in the experiment was a total of 4,288 comments from the buyer's commentary information crawled on the Taobao. For the feature words obtained by the algorithm before and after the improvement, the above program is run to obtain the precision and error rate of the algorithm for the two-pole annotation before and after the improvement of the number of different comments:

Table 7: Emotional Two-level Labeling Precision and Error Rate of Test Sets under Different Number of Comments

Algorithm	comments / index	100 (%)	200 (%)	300 (%)	400 (%)	500 (%)
TF-IDF-NL	Positive precision	85.50	86.23	84.23	89.70	90.33
	Negative precision	84.23	87.33	89.50	90.88	92.45
	Positive error rate	14.50	13.77	15.77	10.30	9.67
	Negative error rate	15.77	12.67	9.50	9.12	7.55
TF-IDF	Positive precision	82.50	79.33	74.00	80.20	81.33
	Negative precision	70.33	77.40	79.30	70.60	71.45
	Positive error rate	17.50	20.67	26.00	19.80	18.67
	Negative error rate	29.67	22.60	20.70	19.40	28.55

For a sentence of emotion, mainly focused on nouns and adjectives, the improved TF-IDF-NL algorithm gives the higher weight of nouns and adjectives in the sentence by setting the weight ratio of part of speech in the sentence. Words are more conducive to later emotional analysis, which is well reflected in Table 7. In addition, the observation table can be found that, within a certain range, the number of comments in the test set is increased, and the precision of the two-level annotation of the emotion is continuously improved, because each time the emotional two-level annotation of a word is completed, It is judged whether the generated comment tuple exists in the training set, and if it does not exist, it is expanded into the training set tuple, which can increase the coverage of the multi-group in the training set to a certain extent.

5. Conclution

This paper puts forward some suggestions for improvement based on the conclusions and methods of the previous TF-IDF algorithm. The main improvements are:

- Assigning feature words of different positions to different weights and highlighting important feature words;
- Realizing the part of speech of feature words as the basis for weight calculation;

The above points are well reflected in the program design, and the results of the improvement are also well displayed, which verifies the feasibility of the improved scheme.

6. References

[1] Ibrahim Abu El-Khair, M.2018. TF*IDF. Encyclopedia of Database Systems (2nd ed.)

[2] Yonghe Lu,Yanfeng Li, J.2013. Improved TF-IDF algorithm for text feature item weight calculation method. Library and Information Service.57(03),90-95.

[3] Lizhen Liu, Yitao Song, J.2004. Feature Selection in Text Classification. Computer Engineering. 30(04),14-15(2004)

[4] Qingyun Yao, Gongshen Liu, Xiang Li, J.2008. Text clustering algorithm based on vector space model. Computer Engineering.18:39-41

[5] Forman G, C.2008. BNS Feature Scaling: An Improved Representation over tf-idf for SVM Text Classification. In: Proceedings of the 17th ACM Conference on Information and Knowledge Management. ACM: 263－270